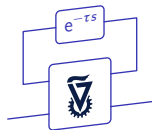


Time Delays in Control and Estimation (038806)



lecture no. 9

Leonid Mirkin

Faculty of Mechanical Engineering
Technion—IIT

Outline

Use of Lyapunov–Krasovskii functionals

Implementation of FIR dead-time compensation elements

Reset mechanism

Finite-dimensional approximations

Lumped-delay approximation

Outline

Use of Lyapunov–Krasovskii functionals

Implementation of FIR dead-time compensation elements

Reset mechanism

Finite-dimensional approximations

Lumped-delay approximation

Lyapunov–Krasovskii functions in delay margin analysis

Consider again

$$\dot{x}(t) = Ax(t) + A_\tau x(t - \tau), \quad \check{x}_0 = 0$$

We seek for conditions under which it is stable $\forall \tau \in [0, \bar{\tau}]$. Typical steps:

1. transform it into a Lyapunov–Krasovskii-friendly form maybe obsolete
2. choose a structure of the Lyapunov–Krasovskii functional an art
3. analyze its derivative via upper-bounding cross terms requires skills

Lyapunov–Krasovskii usage example

More Lyapunov–Krasovskii-friendly model:

$$\begin{aligned}\dot{x}(t) &= (A + A_\tau)x(t) - A_\tau(x(t) - x(t - \tau)) \\ &= (A + A_\tau)x(t) - A_\tau \int_{-\tau}^0 \dot{x}(t + s) ds.\end{aligned}$$

Equivalently, can be written as

$$\underbrace{\begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix}}_{\tilde{E}} \underbrace{\begin{bmatrix} \dot{x}(t) \\ \dot{y}(t) \end{bmatrix}}_{\dot{\tilde{x}}(t)} = \underbrace{\begin{bmatrix} 0 & I \\ A + A_\tau & -I \end{bmatrix}}_{\tilde{A}} \underbrace{\begin{bmatrix} x(t) \\ y(t) \end{bmatrix}}_{\tilde{x}(t)} - \underbrace{\begin{bmatrix} 0 \\ A_\tau \end{bmatrix}}_{\tilde{B}} \int_{-h}^0 y(t + \theta) d\theta$$

by adding the auxiliary variable

$$\dot{x}(t) =: y(t) = \underbrace{\begin{bmatrix} 0 & I \end{bmatrix}}_{\tilde{C}} \tilde{x}(t).$$

Lyapunov–Krasovskii usage example (contd)

Lyapunov–Krasovskii function:

$$V(t) = V_1(t) + V_2(t) := \tilde{x}'(t)P'\tilde{E}\tilde{x}(t) + \int_{-\tau}^0 \int_{t+s}^t y'(r)Ry(r)drds$$

for

$$R > 0 \quad \text{and} \quad P = \begin{bmatrix} P_1 & 0 \\ P_2 & P_3 \end{bmatrix}, \quad P_1 > 0.$$

for which $\tilde{x}'(t)P'\tilde{E}\tilde{x}(t) = x'(t)P_1x(t)$, so that

$$V(\tilde{x}_t) \geq \alpha(\|x(t)\|) \quad \text{for } \alpha(\gamma) = \lambda_{\min}(P_1)\gamma^2.$$

Lyapunov–Krasovskii usage example (contd)

Now, calculate derivatives:

$$\begin{aligned}\dot{V}_1(t) &= 2\tilde{x}'(t)P'\tilde{E}\dot{\tilde{x}}(t) = 2\tilde{x}'(t)P'\left(\tilde{A}\tilde{x}(t) - \tilde{B}\int_{-\tau}^0 y(t+s)ds\right) \\ &= \tilde{x}'(t)(P'\tilde{A} + \tilde{A}'P)\tilde{x}(t) - 2\int_{-\tau}^0 \tilde{x}'(t)P'\tilde{B}y(t+s)ds\end{aligned}$$

and, using the Leibniz integral rule,

$$\begin{aligned}\dot{V}_2(t) &= \int_{-\tau}^0 (y'(t)Ry(t) - y'(t+s)Ry(t+s)) ds \\ &= \tau\tilde{x}'(t)\tilde{C}'R\tilde{C}\tilde{x}(t) - \int_{-\tau}^0 y'(t+s)Ry(t+s)ds.\end{aligned}$$

Because $\tau \leq \bar{\tau}$,

$$\begin{aligned}V(t) &\leq \tilde{x}'(t)(P'\tilde{A} + \tilde{A}'P + \bar{\tau}\tilde{C}'R\tilde{C})\tilde{x}(t) \\ &\quad - \int_{-\tau}^0 y'(t+s)Ry(t+s)ds - 2\int_{-\tau}^0 \tilde{x}'(t)P'\tilde{B}y(t+s)ds.\end{aligned}$$

Lyapunov–Krasovskii usage example (contd)

Now, calculate derivatives:

$$\begin{aligned}\dot{V}_1(t) &= 2\tilde{x}'(t)P'\tilde{E}\dot{\tilde{x}}(t) = 2\tilde{x}'(t)P'\left(\tilde{A}\tilde{x}(t) - \tilde{B}\int_{-\tau}^0 y(t+s)ds\right) \\ &= \tilde{x}'(t)(P'\tilde{A} + \tilde{A}'P)\tilde{x}(t) - 2\int_{-\tau}^0 \tilde{x}'(t)P'\tilde{B}y(t+s)ds\end{aligned}$$

and, using the Leibniz integral rule,

$$\begin{aligned}\dot{V}_2(t) &= \int_{-\tau}^0 (y'(t)Ry(t) - y'(t+s)Ry(t+s)) ds \\ &= \tau\tilde{x}'(t)\tilde{C}'R\tilde{C}\tilde{x}(t) - \int_{-\tau}^0 y'(t+s)Ry(t+s)ds.\end{aligned}$$

Because $\tau \leq \bar{\tau}$,

$$\begin{aligned}\dot{V}(t) &\leq \tilde{x}'(t)(P'\tilde{A} + \tilde{A}'P + \bar{\tau}\tilde{C}'R\tilde{C})\tilde{x}(t) \\ &\quad - \int_{-\tau}^0 y'(t+s)Ry(t+s)ds - 2\int_{-\tau}^0 \tilde{x}'(t)P'\tilde{B}y(t+s)ds.\end{aligned}$$

Technical result

For all $Q > 0$ and vectors v_1 and v_2 , we have that

$$0 \leq (v_1 + Q^{-1}v_2)'Q(v_1 + Q^{-1}v_2) = v_1'Qv_1 + v_2'Q^{-1}v_2 + 2v_2'v_1.$$

Thus,

$$-2v_2'v_1 \leq v_1'Qv_1 + v_2'Q^{-1}v_2.$$

Hence,

$$\xi := 2 \int_{-\tau}^0 \dot{x}'(t)P'\tilde{B}y(t+s)ds$$

satisfies

$$\begin{aligned} -\xi &\leq \int_{-\tau}^0 \left(\dot{x}'(t)P'\tilde{B}Q^{-1}\tilde{B}'P\dot{x}'(t) + y'(t+s)Qy(t+s) \right) ds \\ &\leq \tau \dot{x}'(t)P'\tilde{B}Q^{-1}\tilde{B}'P\dot{x}'(t) + \int_{-\tau}^0 y'(t+s)Qy(t+s)ds, \end{aligned}$$

for all $Q > 0$ (including $Q = R$).

Technical result

For all $Q > 0$ and vectors v_1 and v_2 , we have that

$$0 \leq (v_1 + Q^{-1}v_2)'Q(v_1 + Q^{-1}v_2) = v_1'Qv_1 + v_2'Q^{-1}v_2 + 2v_2'v_1.$$

Thus,

$$-2v_2'v_1 \leq v_1'Qv_1 + v_2'Q^{-1}v_2.$$

Hence,

$$\xi := 2 \int_{-\tau}^0 \tilde{x}'(t)P'\tilde{B}y(t+s)ds$$

satisfies

$$\begin{aligned} -\xi &\leq \int_{-\tau}^0 \left(\tilde{x}'(t)P'\tilde{B}Q^{-1}\tilde{B}'P\tilde{x}'(t) + y'(t+s)Qy(t+s) \right) ds \\ &\leq \bar{\tau}\tilde{x}'(t)P'\tilde{B}Q^{-1}\tilde{B}'P\tilde{x}'(t) + \int_{-\tau}^0 y'(t+s)Qy(t+s)ds, \end{aligned}$$

for all $Q > 0$ (including $Q = R$).

Lyapunov–Krasovskii usage example (contd)

Thus,

$$\begin{aligned}
 \dot{V}(t) &\leq \tilde{x}'(t)(P'\tilde{A} + \tilde{A}'P + \bar{\tau}\tilde{C}'R\tilde{C})\tilde{x}(t) \\
 &\quad - \int_{-\tau}^0 y'(t+s)Ry(t+s)ds - 2 \int_{-\tau}^0 \tilde{x}'(t)P'\tilde{B}y(t+s)ds \\
 &\leq \tilde{x}'(t)(P'\tilde{A} + \tilde{A}'P + \bar{\tau}\tilde{C}'R\tilde{C})\tilde{x}(t) - \int_{-\tau}^0 y'(t+s)Ry(t+s)ds \\
 &\quad - \bar{\tau}\tilde{x}'(t)P'\tilde{B}R^{-1}\tilde{B}'P\tilde{x}(t) + \int_{-\tau}^0 y'(t+s)Ry(t+s)ds \\
 &\leq \tilde{x}'(t)(P'\tilde{A} + \tilde{A}'P + \bar{\tau}\tilde{C}'R\tilde{C} + \bar{\tau}P'\tilde{B}R^{-1}\tilde{B}'P)\tilde{x}(t)
 \end{aligned}$$

Thus, $\dot{V}(t) < 0$ for all $\tilde{x} \neq 0$ if

$$P'\tilde{A} + \tilde{A}'P + \bar{\tau}\tilde{C}'R\tilde{C} + \bar{\tau}P'\tilde{B}R^{-1}\tilde{B}'P < 0.$$

Lyapunov–Krasovskii usage example (contd)

Thus,

$$\begin{aligned}
 \dot{V}(t) &\leq \tilde{x}'(t)(P'\tilde{A} + \tilde{A}'P + \bar{\tau}\tilde{C}'R\tilde{C})\tilde{x}(t) \\
 &\quad - \int_{-\tau}^0 y'(t+s)Ry(t+s)ds - 2 \int_{-\tau}^0 \tilde{x}'(t)P'\tilde{B}y(t+s)ds \\
 &\leq \tilde{x}'(t)(P'\tilde{A} + \tilde{A}'P + \bar{\tau}\tilde{C}'R\tilde{C})\tilde{x}(t) - \int_{-\tau}^0 y'(t+s)Ry(t+s)ds \\
 &\quad - \bar{\tau}\tilde{x}'(t)P'\tilde{B}R^{-1}\tilde{B}'P\tilde{x}(t) + \int_{-\tau}^0 y'(t+s)Ry(t+s)ds \\
 &\leq \tilde{x}'(t)(P'\tilde{A} + \tilde{A}'P + \bar{\tau}\tilde{C}'R\tilde{C} + \bar{\tau}P'\tilde{B}R^{-1}\tilde{B}'P)\tilde{x}(t)
 \end{aligned}$$

Thus, $\dot{V}(t) < 0$ for all $\tilde{x} \neq 0$ if

$$P'\tilde{A} + \tilde{A}'P + \bar{\tau}\tilde{C}'R\tilde{C} + \bar{\tau}P'\tilde{B}R^{-1}\tilde{B}'P < 0.$$

Lyapunov–Krasovskii usage example (contd)

Using the Schur complement technique we end up with

$$\begin{bmatrix} P'\tilde{A} + \tilde{A}'P + \bar{\tau}\tilde{C}'R\tilde{C} & \bar{\tau}P'\tilde{B} \\ \bar{\tau}\tilde{B}'P & -\bar{\tau}R \end{bmatrix} < 0,$$

which is an LMI, efficiently solvable in P and R for each $\bar{\tau}$.

In general,

- many approaches to reduce conservatism
chiefly at the expense of computational complexity
- not design friendly

Lyapunov–Krasovskii usage example (contd)

Using the Schur complement technique we end up with

$$\begin{bmatrix} P'\tilde{A} + \tilde{A}'P + \bar{\tau}\tilde{C}'R\tilde{C} & \bar{\tau}P'\tilde{B} \\ \bar{\tau}\tilde{B}'P & -\bar{\tau}R \end{bmatrix} < 0,$$

which is an LMI, efficiently solvable in P and R for each $\bar{\tau}$.

Remark: The same LMI by applying the embedding approach to $P(s) = (sI - A)^{-1}A_\tau$, $W_\kappa(s) = 1$ and $W_\alpha(s) = \bar{\tau}s$, for which

$$T_{\text{rob}}(s) = \bar{\tau}s(sI - A - A_\tau)^{-1}A_\tau = \bar{\tau} \begin{bmatrix} 0 & I \end{bmatrix} \left(s \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} - \begin{bmatrix} 0 & I \\ A + A_\tau & -I \end{bmatrix} \right)^{-1} \begin{bmatrix} 0 \\ A_\tau \end{bmatrix}$$

which is not economical and easy to improve (e.g. $W_{\alpha 1,0} \rightarrow W_{\alpha 1,1}$).

In general,

- many approaches to reduce conservatism chiefly at the expense of computational complexity
- not design friendly

Lyapunov–Krasovskii usage example (contd)

Using the Schur complement technique we end up with

$$\begin{bmatrix} P'\tilde{A} + \tilde{A}'P + \bar{\tau}\tilde{C}'R\tilde{C} & \bar{\tau}P'\tilde{B} \\ \bar{\tau}\tilde{B}'P & -\bar{\tau}R \end{bmatrix} < 0,$$

which is an LMI, efficiently solvable in P and R for each $\bar{\tau}$.

Remark: The same LMI by applying the embedding approach to $P(s) = (sI - A)^{-1}A_\tau$, $W_\kappa(s) = 1$ and $W_\alpha(s) = \bar{\tau}s$, for which

$$T_{\text{rob}}(s) = \bar{\tau}s(sI - A - A_\tau)^{-1}A_\tau = \bar{\tau} \begin{bmatrix} 0 & I \end{bmatrix} \left(s \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} - \begin{bmatrix} 0 & I \\ A + A_\tau & -I \end{bmatrix} \right)^{-1} \begin{bmatrix} 0 \\ A_\tau \end{bmatrix}$$

which is not economical and easy to improve (e.g. $W_{\alpha 1,0} \rightarrow W_{\alpha 1,1}$).

In general,

- ☺ many approaches to reduce conservatism
chiefly at the expense of computational complexity
- ☹ not design friendly

Outline

Use of Lyapunov–Krasovskii functionals

Implementation of FIR dead-time compensation elements

Reset mechanism

Finite-dimensional approximations

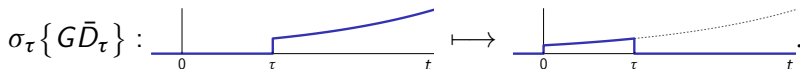
Lumped-delay approximation

FIR DTC element

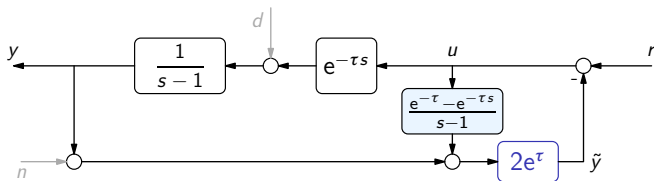
DTC for systems with a single delay is of the form

$$\begin{aligned}
 \Pi(s) &= \sigma_{\tau} \left\{ \left[\begin{array}{c|c} A & B \\ \hline C & 0 \end{array} \right] e^{-\tau s} \right\} = \left[\begin{array}{c|c} A & e^{-A\tau} B \\ \hline C & 0 \end{array} \right] - \left[\begin{array}{c|c} A & B \\ \hline C & 0 \end{array} \right] e^{-\tau s} \\
 &= C(sI - A)^{-1} e^{-A\tau} B - C(sI - A)^{-1} B e^{-\tau s} \\
 &= C(sI - A)^{-1} \begin{bmatrix} e^{-A\tau} B & B \end{bmatrix} \begin{bmatrix} I \\ e^{-\tau s} I \end{bmatrix} \\
 &= \left[\begin{array}{c|c|c} A & e^{-A\tau} B & B \\ \hline C & 0 & 0 \end{array} \right] \begin{bmatrix} I \\ e^{-\tau s} I \end{bmatrix} \\
 &= C \int_0^{\tau} e^{-(sI - A)t} dt e^{-A\tau} B
 \end{aligned}$$

where the completion operator



Benchmark example



has

$$T_{yr}(s) = \frac{e^{-\tau s}}{s+1}.$$

How could we implement the DTC element

$$\Pi(s) = \sigma_{\tau} \left\{ \frac{e^{-\tau s}}{s-1} \right\} ?$$

“Natural” implementation

as

$$\begin{cases} \dot{x}_n(t) = Ax_n(t) + e^{-A\tau}Bu(t) - Bu(t - \tau) \\ y_n(t) = Cx_n(t) \end{cases}$$

should result in

$$y_n(t) = C \int_0^t e^{A(t-r)} (e^{-A\tau}Bu(r) - Bu(r - \tau)) dr = C \int_{t-\tau}^t e^{A(t-r-\tau)} Bu(r) dr$$

and closed-loop step response $y(t) = (1 - e^{-(t-1)})\mathbb{1}(t-1)$ but results in

“Natural” implementation

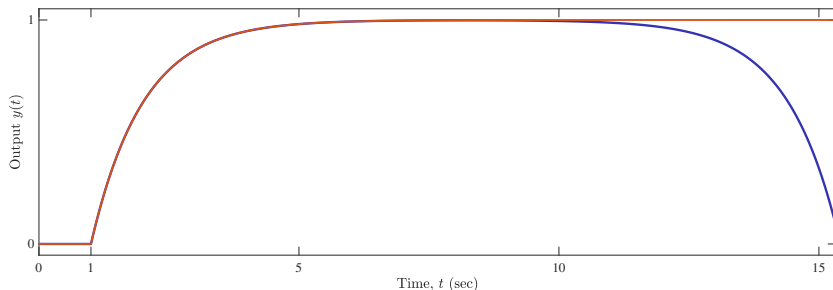
as

$$\begin{cases} \dot{x}_n(t) = Ax_n(t) + e^{-A\tau}Bu(t) - Bu(t - \tau) \\ y_n(t) = Cx_n(t) \end{cases}$$

should result in

$$y_n(t) = C \int_0^t e^{A(t-r)} (e^{-A\tau}Bu(r) - Bu(r - \tau)) dr = C \int_{t-\tau}^t e^{A(t-r-\tau)} Bu(r) dr$$

and closed-loop step response $y(t) = (1 - e^{-(t-1)})\mathbb{1}(t-1)$, but results in



“Natural” implementation (contd)

More realistically,

$$\begin{cases} \dot{x}_n(t) = Ax_n(t) + e^{-A\tau}Bu(t) - Bu(t - \tau) + \epsilon_{\text{roff}}(t) \\ y_n(t) = Cx_n(t) \end{cases}$$

Solution:

$$\begin{aligned} \tilde{y}_n(t) &= C \int_0^t e^{A(t-s)} (e^{-A\tau}Bu(s) - Bu(s - \tau) + \epsilon_{\text{roff}}(s)) ds \\ &= C \underbrace{\int_{t-\tau}^t e^{A(t-s-\tau)} Bu(s) ds}_{y_n(t)} + C \underbrace{\int_0^t e^{A(t-s)} \epsilon_{\text{roff}}(s) ds}_{\epsilon_n(t)} \end{aligned}$$

and ϵ_n might **diverge if A is not Hurwitz**.

Conclusion:

→ we need something more sophisticated to implement Π .

“Natural” implementation (contd)

More realistically,

$$\begin{cases} \dot{x}_\Pi(t) = Ax_\Pi(t) + e^{-A\tau}Bu(t) - Bu(t - \tau) + \epsilon_{\text{roff}}(t) \\ y_\Pi(t) = Cx_\Pi(t) \end{cases}$$

Solution:

$$\begin{aligned} \tilde{y}_\Pi(t) &= C \int_0^t e^{A(t-s)} (e^{-A\tau}Bu(s) - Bu(s - \tau) + \epsilon_{\text{roff}}(s)) ds \\ &= C \underbrace{\int_{t-\tau}^t e^{A(t-s-\tau)} Bu(s) ds}_{y_\Pi(t)} + C \underbrace{\int_0^t e^{A(t-s)} \epsilon_{\text{roff}}(s) ds}_{\epsilon_\Pi(t)} \end{aligned}$$

and ϵ_Π might **diverge if A is not Hurwitz**.

Conclusion:

- we need something more sophisticated to implement $\Pi \dots$

Outline

Use of Lyapunov–Krasovskii functionals

Implementation of FIR dead-time compensation elements

Reset mechanism

Finite-dimensional approximations

Lumped-delay approximation

Underlying idea

The impulse response of Π ,

$$\pi(t) = Ce^{A(t-\tau)}B\mathbb{1}_{[0,\tau]}(t)$$

has support in $[0, \tau]$, so that for any t_c

- Π “forgets” its initial condition $(x_\Pi(t_c), \check{u}_{t_c})$ after $t = t_c + \tau$.

Hence we may

1. reset the state of Π once in a while,
2. wait τ time units to accumulate the history of inputs
3. use the output for a while after that (while the error ϵ_Π is not “large”)

While we wait for a wrong initial state to wipe out of the memory, another output shall be used $\implies$ we need more than 1 system to run.

Reset Π

If the whole state $(x_\Pi(t), \check{u}_t)$ is reset to zero at t_c , then for each $t > t_c$

$$\begin{aligned}
 x_\Pi(t) &= \int_{t_c}^t e^{A(t-s)} (e^{-A\tau} Bu(s) - Bu(s-\tau) \mathbb{1}(s-t_c-\tau) + \epsilon_{\text{roff}}(s)) ds \\
 &= \int_{t_c}^t e^{A(t-s-\tau)} Bu(s) ds - \int_{\min\{t_c+\tau, t\}}^t e^{A(t-s)} Bu(s-\tau) ds + \epsilon_\Pi(t) \\
 &= \int_{t_c}^t e^{A(t-s-\tau)} Bu(s) ds - \int_{\min\{t_c, t-\tau\}}^{t-\tau} e^{A(t-s-\tau)} Bu(s) ds + \epsilon_\Pi(t) \\
 &= \int_{t-\tau}^t e^{A(t-s-\tau)} Bu(s) ds - \int_{\min\{t_c, t-\tau\}}^{t_c} e^{A(t-s-\tau)} Bu(s) ds + \epsilon_\Pi(t)
 \end{aligned}$$

Hence,

$$\tilde{y}_\Pi(t) = y_\Pi(t) - \overbrace{C \int_{\min\{t_c, t-\tau\}}^{t_c} e^{A(t-s-\tau)} Bu(s) ds}^{y_\delta(t)} + C\epsilon_\Pi(t)$$

with y_δ vanishing after $t = t_c + \tau$.

Roundoff error

If reset every T time units,

$$\epsilon_{\text{r}}(t) = \int_{t_c}^t e^{A(t-s)} \epsilon_{\text{roff}}(s) ds$$

remains bounded, with

$$\|\epsilon_{\text{r}}\|_{\infty} \leq \int_0^T \|C e^{At}\|_{\infty} dt \|\epsilon_{\text{roff}}\|_{\infty} < \infty$$

Implementation

Π :

$$\begin{cases} \dot{x}_1(t) = Ax_1(t) + e^{-A\tau}Bu(t) - \sigma_t Bu(t - \tau), & x_1((2k+1)\tau) = 0 \\ \dot{x}_2(t) = Ax_2(t) + e^{-A\tau}Bu(t) - (1 - \sigma_t)Bu(t - \tau), & x_2(2k\tau) = 0 \\ y_\Pi(t) = \sigma_t Cx_1(t) + (1 - \sigma_t)Cx_2(t) + Du(t - \tau) \end{cases}$$

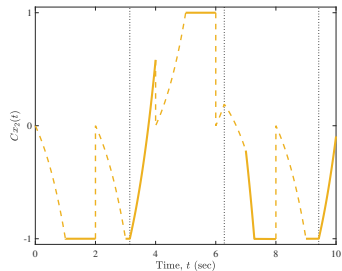
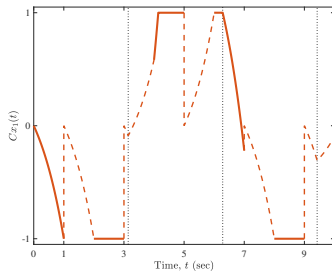
where

$$\sigma_t = \begin{cases} 1 & \text{if } t \in [2k\tau, (2k+1)\tau) \\ 0 & \text{if } t \in [(2k+1)\tau, (2k+2)\tau) \end{cases}$$

for all $k \in \mathbb{Z}_+$.

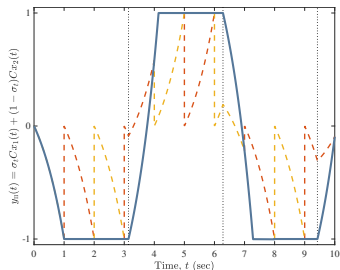
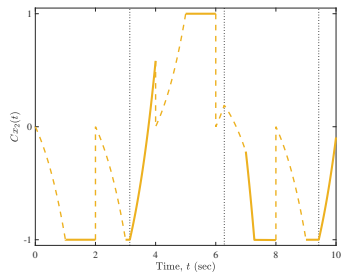
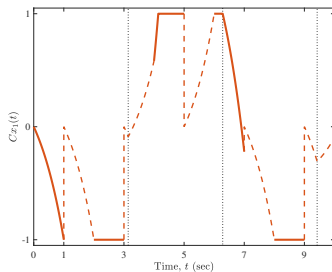
Implementation

The response to a square wave:



Implementation

The response to a square wave:



Pros and cons

- 😊 easy to implement
- ☹ time-varying system $\implies$ harder to analyze accuracy
- ☹ runs 2τ between resets, problematic if unstable modes of A are faster
if implement ν systems, then runs $\tau\nu/(\nu - 1) > \tau$ between resets

Outline

Use of Lyapunov–Krasovskii functionals

Implementation of FIR dead-time compensation elements

Reset mechanism

Finite-dimensional approximations

Lumped-delay approximation

Naïve Padé

In

$$\Pi(s) = \left[\frac{A}{C} \middle| \frac{e^{-A\tau} B}{0} \right] - \left[\frac{A}{C} \middle| \frac{B}{0} \right] e^{-\tau s}$$

substitute $e^{-\tau s} \rightarrow R_{n,n}(s)$.

☹ manual cancellation of unstable poles required (problematic in MIMO).

Still,

☺ requires only standard tools

In our example,

$$\Pi_{\text{naiveP},n}(s) = \begin{cases} \frac{0.632121(s-1.00146)(s-11.9822)}{(s-1)(s^2+6s+12)} & n=2 \\ \frac{1.36788(s-0.99998969)(s^2-4.54542s+55.4546)}{(s-1)(s+4.64437)(s^2+7.35563s+25.8377)} & n=3 \\ -\frac{0.6321(s-1.000000004)(s-39.8821)(s^2-2.39698s+42.1242)}{(s-1)(s^2+8.41516s+45.9512)(s^2+11.5848s+36.5605)))} & n=4 \end{cases}$$

Naïve Padé

In

$$\Pi(s) = \left[\frac{A}{C} \middle| \frac{e^{-A\tau} B}{0} \right] - \left[\frac{A}{C} \middle| \frac{B}{0} \right] e^{-\tau s}$$

substitute $e^{-\tau s} \rightarrow R_{n,n}(s)$.

☹ manual cancellation of unstable poles required (problematic in MIMO).

Still,

☺ requires only standard tools

In our example,

$$\Pi_{\text{naïveP},n}(s) = \begin{cases} -\frac{0.632121(s-\mathbf{1.00148})(s-11.9822)}{(s-1)(s^2+6s+12)} & n=2 \\ \frac{1.36788(s-\mathbf{0.99998969})(s^2-4.54542s+55.4546)}{(s-1)(s+4.64437)(s^2+7.35563s+25.8377)} & n=3 \\ -\frac{0.6321(s-\mathbf{1.00000004})(s-39.8821)(s^2-2.39698s+42.1242)}{(s-1)(s^2+8.41516s+45.9512)(s^2+11.5848s+36.5605))} & n=4 \end{cases}$$

Interpolating Padé

In

$$\Pi(s) = \left[\frac{A}{C} \middle| \frac{e^{-A\tau} B}{0} \right] - \left[\frac{A}{C} \middle| \frac{B}{0} \right] e^{-\tau s}$$

substitute $e^{-\tau s} \rightarrow \tilde{R}_{n,n}(s)$ such that

$$e^{-\tau \lambda_i} = \tilde{R}_{n,n}(\lambda_i)$$

for all unstable eigenvalues of A . Consequently,

- ☺ unstable modes still canceled
- ☹ might be problematic numerically
- ☹ less Maclaurin series terms matched

In our example,

$$\Pi_{\text{intP},n}(s) = \begin{cases} \frac{0.451471(s-15.26403)}{s^2+5.6387s+10.9161} & n=2 \\ \frac{1.23504(s^2-4.29253s+57.3393)}{(s+4.51376)(s^2+7.08771s+24.8196)} & n=3 \\ \frac{0.527108(s-45.0554)(s^2-2.36514s+42.3681)}{(s^2+8.20247s+45.0126)(s^2+11.3775s+35.3632)} & n=4 \end{cases}$$

Interpolating Padé

In

$$\Pi(s) = \left[\frac{A}{C} \middle| \frac{e^{-A\tau} B}{0} \right] - \left[\frac{A}{C} \middle| \frac{B}{0} \right] e^{-\tau s}$$

substitute $e^{-\tau s} \rightarrow \tilde{R}_{n,n}(s)$ such that

$$e^{-\tau \lambda_i} = \tilde{R}_{n,n}(\lambda_i)$$

for all unstable eigenvalues of A . Consequently,

- ☺ unstable modes still canceled
- ☹ might be problematic numerically
- ☹ less Maclaurin series terms matched

In our example,

$$\Pi_{\text{intP},n}(s) = \begin{cases} -\frac{0.451471(s-15.28403)}{s^2+5.6387s+10.9161} & n=2 \\ \frac{1.23504(s^2-4.29253s+57.3393)}{(s+4.51376)(s^2+7.08771s+24.8196)} & n=3 \\ -\frac{0.527108(s-45.0554)(s^2-2.36514s+42.3681)}{(s^2+8.20247s+45.0126)(s^2+11.3775s+35.3632)} & n=4 \end{cases}$$

Partington–Mäkilä

Rewrite

$$\Pi(s) = C\phi([sI - A])e^{-A\tau}B, \quad \text{where } \phi(s) := \frac{1 - e^{-\tau s}}{s} = \pi_\tau \left\{ \frac{1}{s} \right\}.$$

Use

$$\Pi(s) \approx CR_\phi([sI - A])e^{-A\tau}B$$

for any rational approximation $R_\phi(s)$ of $\phi(s)$.

- ☺ easy to approximate $\phi(s)$
- ☹ not always results in a stable approximation
- ☹ dimensions increase

In our example,

$$\Pi_{\text{P.M.},n}(s) = \begin{cases} \frac{4.41455}{s^2 + 4s + 7} & n = 2 \\ \frac{0.735759(s^2 - 2s + 61)}{(s + 3.64437)(s^2 + 5.35563s + 19.4821)} & n = 3 \\ \frac{14.7152(s^2 - 2s + 43)}{(s^2 + 6.41516s + 38.536)(s^2 + 9.58484s + 25.9757)} & n = 4 \end{cases}$$

Partington–Mäkilä

Rewrite

$$\Pi(s) = C\phi([sI - A])e^{-A\tau}B, \quad \text{where } \phi(s) := \frac{1 - e^{-\tau s}}{s} = \pi_\tau \left\{ \frac{1}{s} \right\}.$$

Use

$$\Pi(s) \approx CR_\phi([sI - A])e^{-A\tau}B$$

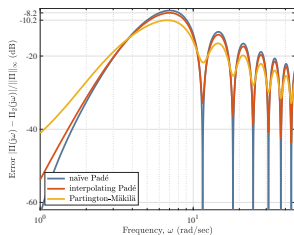
for any rational approximation $R_\phi(s)$ of $\phi(s)$.

- ☺ easy to approximate $\phi(s)$
- ☹ not always results in a stable approximation
- ☹ dimensions increase

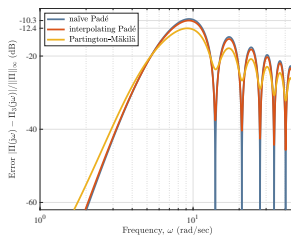
In our example,

$$\Pi_{\text{PaMä},n}(s) = \begin{cases} \frac{4.41455}{s^2+4s+7} & n = 2 \\ \frac{0.735759(s^2-2s+61)}{(s+3.64437)(s^2+5.35563s+19.4821)} & n = 3 \\ \frac{14.7152(s^2-2s+43)}{(s^2+6.41516s+38.536)(s^2+9.58484s+25.9757)} & n = 4 \end{cases}$$

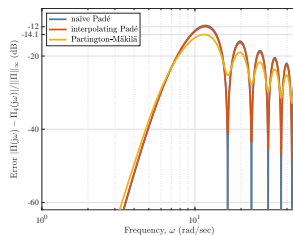
Comparisons



$n = 2$



$n = 3$



$n = 4$

Outline

Use of Lyapunov–Krasovskii functionals

Implementation of FIR dead-time compensation elements

Reset mechanism

Finite-dimensional approximations

Lumped-delay approximation

Newton–Cotes formulae

$$\int_a^b f(t)dt \approx \sum_{i=0}^v \phi_i f(t_i), \quad \text{where } t_i := a + i(b-a)/v$$

where Φ_i are determined by approximation method.

Trapezoidal:

$$\Rightarrow \quad \phi_i = \frac{b-a}{2v} \begin{cases} 2 & \text{if } i = 1, \dots, v-1 \\ 1 & \text{if } i = 0, v \end{cases}$$

Simpson's:

$$\Rightarrow \quad \phi_i = \frac{b-a}{6v} \begin{cases} 1 & \text{if } i = 0, v \\ 2 & \text{if } i \text{ even} \\ 4 & \text{if } i \text{ odd} \end{cases}$$

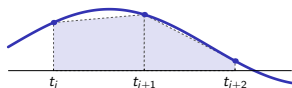
All work if derivatives of f are bounded.

Newton–Cotes formulae

$$\int_a^b f(t)dt \approx \sum_{i=0}^v \phi_i f(t_i), \quad \text{where } t_i := a + i(b-a)/v$$

where Φ_i are determined by approximation method.

Trapezoidal:



$$\Rightarrow \phi_i = \frac{b-a}{2v} \begin{cases} 2 & \text{if } i = 1, \dots, v-1 \\ 1 & \text{if } i = 0, v \end{cases}$$

Simpson's:

$$\Rightarrow \phi_i = \frac{b-a}{6v} \begin{cases} 1 & \text{if } i = 0, v \\ 2 & \text{if } i \text{ even} \\ 4 & \text{if } i \text{ odd} \end{cases}$$

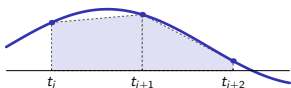
All work if derivatives of f are bounded.

Newton–Cotes formulae

$$\int_a^b f(t)dt \approx \sum_{i=0}^v \phi_i f(t_i), \quad \text{where } t_i := a + i(b-a)/v$$

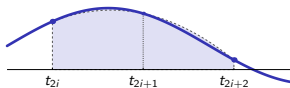
where Φ_i are determined by approximation method.

Trapezoidal:



$$\Rightarrow \phi_i = \frac{b-a}{2v} \begin{cases} 2 & \text{if } i = 1, \dots, v-1 \\ 1 & \text{if } i = 0, v \end{cases}$$

Simpson's:



$$\Rightarrow \phi_i = \frac{b-a}{6v} \begin{cases} 1 & \text{if } i = 0, v \\ 2 & \text{if } i \text{ even} \\ 4 & \text{if } i \text{ odd} \end{cases}$$

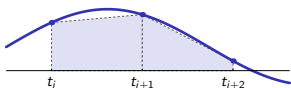
All work if derivatives of f are bounded.

Newton–Cotes formulae

$$\int_a^b f(t)dt \approx \sum_{i=0}^v \phi_i f(t_i), \quad \text{where } t_i := a + i(b-a)/v$$

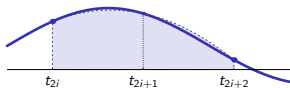
where Φ_i are determined by approximation method.

Trapezoidal:



$$\Rightarrow \phi_i = \frac{b-a}{2v} \begin{cases} 2 & \text{if } i = 1, \dots, v-1 \\ 1 & \text{if } i = 0, v \end{cases}$$

Simpson's:



$$\Rightarrow \phi_i = \frac{b-a}{6v} \begin{cases} 1 & \text{if } i = 0, v \\ 2 & \text{if } i \text{ even} \\ 4 & \text{if } i \text{ odd} \end{cases}$$

All work if derivatives of f are bounded.

Naïve Newton–Cotes

If

$$\Pi(s) = C \int_0^\tau e^{-(sI-A)t} dt e^{-A\tau} B = \int_0^\tau C e^{A(t-\tau)} B e^{-ts} dt,$$

then

$$f(t) = C e^{A(t-\tau)} B e^{-ts}.$$

We may try then

$$\Pi(s) \approx \Pi_\nu(s) := \sum_{i=0}^{\nu} \phi_i C e^{A(\tau_i-\tau)} B e^{-\tau_i s}, \quad \text{where } \tau_i := \frac{\tau}{\nu} i.$$

which is called a **lumped-delay approximation** (LDA) of Π . Clearly,

$$\Pi_\nu \in H_\infty$$

whenever ϕ_i are finite

Naïve Newton–Cotes

If

$$\Pi(s) = C \int_0^\tau e^{-(sI-A)t} dt e^{-A\tau} B = \int_0^\tau C e^{A(t-\tau)} B e^{-ts} dt,$$

then

$$f(t) = C e^{A(t-\tau)} B e^{-ts}.$$

We may try then

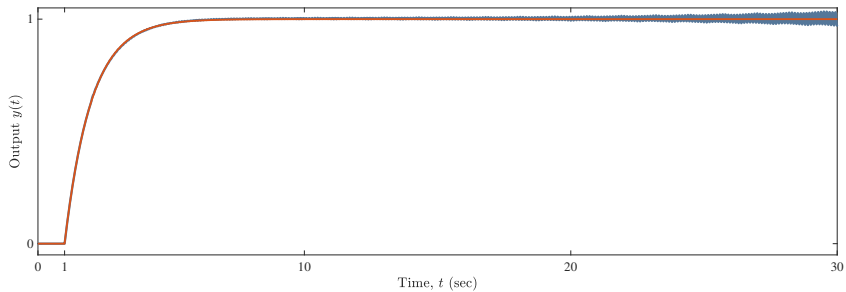
$$\Pi(s) \approx \Pi_\nu(s) := \sum_{i=0}^{\nu} \phi_i C e^{A(\tau_i-\tau)} B e^{-\tau_i s}, \quad \text{where } \tau_i := \frac{\tau}{\nu} i.$$

which is called a **lumped-delay approximation** (LDA) of Π . Clearly,

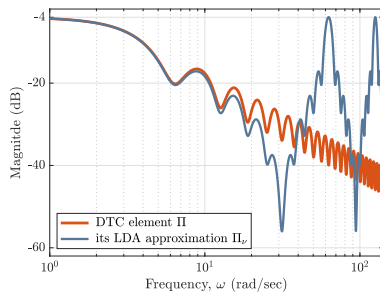
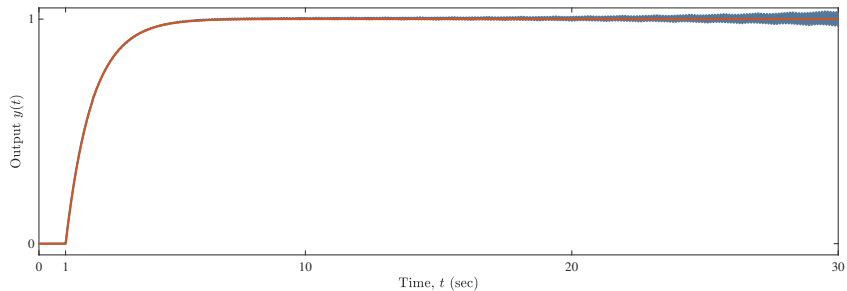
$$\Pi_\nu \in H_\infty$$

whenever ϕ_i are finite

Results



Results



What went wrong?

Because $\Pi, \Pi_v \in H_\infty$, the error

$$\|\Pi - \Pi_v\|_\infty = \operatorname{ess\,sup}_\omega \|\Pi(j\omega) - \Pi_v(j\omega)\|$$

and we need to understand whether

$$f(t) = Ce^{A(t-\tau)}Be^{-j\omega t}$$

is a good integrand to approximate by Newton–Cotes. To this end, consider

$$\dot{f}(t) = Ce^{A(t-\tau)}(A - j\omega I)Be^{-j\omega t},$$

for which

$$\begin{aligned}\|\dot{f}(t)\|_F^2 &= \|Ce^{A(t-\tau)}(A - j\omega I)Be^{-j\omega t}\|_F^2 \\ &= \|Ce^{A(t-\tau)}(A - j\omega I)B\|_F^2 = \|Ce^{A(t-\tau)}AB\|_F^2 + \omega^2 \|Ce^{A(t-\tau)}B\|_F^2\end{aligned}$$

is unbounded:

What went wrong?

Because $\Pi, \Pi_v \in H_\infty$, the error

$$\|\Pi - \Pi_v\|_\infty = \operatorname{ess\,sup}_\omega \|\Pi(j\omega) - \Pi_v(j\omega)\|$$

and we need to understand whether

$$f(t) = Ce^{A(t-\tau)}Be^{-j\omega t}$$

is a good integrand to approximate by Newton–Cotes. To this end, consider

$$\dot{f}(t) = Ce^{A(t-\tau)}(A - j\omega I)Be^{-j\omega t},$$

for which

$$\begin{aligned}\|\dot{f}(t)\|_F^2 &= \|Ce^{A(t-\tau)}(A - j\omega I)Be^{-j\omega t}\|_F^2 \\ &= \|Ce^{A(t-\tau)}(A - j\omega I)B\|_F^2 = \|Ce^{A(t-\tau)}AB\|_F^2 + \omega^2 \|Ce^{A(t-\tau)}B\|_F^2\end{aligned}$$

is unbounded :(

A fix

Integrating by parts,

$$\begin{aligned}
 s \int_0^\tau e^{-(sI-A)t} dt &= - \int_0^\tau e^{At} \left(\frac{d}{dt} e^{-st} \right) dt \\
 &= -e^{-(sI-A)t} \Big|_0^\tau + \int_0^\tau \left(\frac{d}{dt} e^{At} \right) e^{-st} dt \\
 &= I - e^{A\tau} e^{-\tau s} + \int_0^\tau e^{-(sI-A)t} dt A,
 \end{aligned}$$

Hence,

$$\begin{aligned}
 \Pi(s) &= \frac{1}{s+\alpha} (s+\alpha) C \int_0^\tau e^{-(sI-A)t} dt e^{-A\tau} B \\
 &= \frac{1}{s+\alpha} C \left(I - e^{A\tau} e^{-\tau s} + \int_0^\tau e^{-(sI-A)t} dt A + \alpha \int_0^\tau e^{-(sI-A)t} dt \right) e^{-A\tau} B \\
 &= \frac{C e^{-A\tau} B}{s+\alpha} + \frac{1}{s+\alpha} \int_0^\tau C e^{A(t-\tau)} (\alpha I + A) B e^{-ts} dt - \frac{CB}{s+\alpha} e^{-\tau s}
 \end{aligned}$$

for all $\alpha \geq 0$.

A fix

Integrating by parts,

$$\begin{aligned}
 s \int_0^\tau e^{-(sI-A)t} dt &= - \int_0^\tau e^{At} \left(\frac{d}{dt} e^{-st} \right) dt \\
 &= -e^{-(sI-A)t} \Big|_0^\tau + \int_0^\tau \left(\frac{d}{dt} e^{At} \right) e^{-st} dt \\
 &= I - e^{A\tau} e^{-\tau s} + \int_0^\tau e^{-(sI-A)t} dt A,
 \end{aligned}$$

Hence,

$$\begin{aligned}
 \Pi(s) &= \frac{1}{s+\alpha} (s+\alpha) C \int_0^\tau e^{-(sI-A)t} dt e^{-A\tau} B \\
 &= \frac{1}{s+\alpha} C \left(I - e^{A\tau} e^{-\tau s} + \int_0^\tau e^{-(sI-A)t} dt A + \alpha \int_0^\tau e^{-(sI-A)t} dt \right) e^{-A\tau} B \\
 &= \frac{C e^{-A\tau} B}{s+\alpha} + \underbrace{\frac{1}{s+\alpha} \int_0^\tau C e^{A(t-\tau)} (\alpha I + A) B e^{-ts} dt}_{\text{Newton–Cotes}} - \frac{CB}{s+\alpha} e^{-\tau s}
 \end{aligned}$$

for all $\alpha > 0$.

Beyond Newton–Cotes

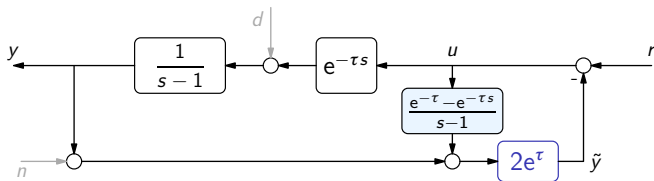
General

$$\Pi_v(s) = F(s) \sum_{i=0}^v \bar{\Pi}_i e^{-\tau_i s}$$

for

- $v \in \mathbb{N}$
- $\{\tau_i\}$ such that $0 = \tau_0 < \tau_1 < \cdots < \tau_{v-1} < \tau_v = \tau$
- $F \in H_\infty$ with strictly proper $F(s)$
- matrix-valued coefficients $\bar{\Pi}_i$, $i \in \mathbb{Z}_{0,v}$.

Example: implementation



with the trapezoidal method,

$$\Pi_{v,\alpha}(s) = \frac{1}{s + \alpha} \left(\frac{(2v + (\alpha + 1)\tau)e^{-\tau}}{2v} + \sum_{i=1}^{v-1} \frac{\tau e^{\tau(i/v-1)}(\alpha + 1)}{v} e^{-i(\tau/v)s} - \frac{2v - (\alpha + 1)\tau}{2v} e^{-\tau s} \right).$$

Note that

$$\Pi_{v,\alpha}(0) = \frac{\tau(1 + \alpha) \coth \frac{\tau}{2v} - 2v}{2\alpha v} (1 - e^{-\tau}) \neq 1 - e^{-\tau}.$$